
title: Run SUAVE Locally description: Start running SUAVE and participating in the chain itself keywords: - build - suave - setup - practice

You can use suave-geth to start a local SUAVE devnet.

There are two ways to work with suave-geth:

1. < Install the latest release binary
2. ♀ Build from source

Latest release binary

```
bash curl -L https://suaveup.flashbots.net | bash
```

Start your local devnet with:

```
bash suave-geth --suave.dev
```

Building from source

:::info

We recommend that you use Golang v1.21 or newer.

:::

Clone the [suave-geth](#) repository:

```
bash git clone git@github.com:flashbots/suave-geth.git cd suave-geth
```

Build the binary:

```
bash make suave
```

Now you have a suave binary in ./build/bin/:

```
bash ./build/bin/suave-geth --version
```

Start the local devnet with:

```
bash ./build/bin/suave-geth --suave.dev
```

► Missing packages

Testing the devnet

Compile the example contracts:

► What is Forge?

```
bash cd suave && forge build
```

Create a few example transactions:

```
bash go run devenv/cmd/main.go
```

Execute a RPC request with curl like this:

```
bash curl 'http://localhost:8545' --header 'Content-Type: application/json' --data '{ "jsonrpc": "2.0", "method": "eth_blockNumber", "params": [], "id": 83 }'
```

If you built from source (but not if you're running Docker), you can attach to the usual Geth javascript console to get any interactive data you need with:

```
bash ./build/bin/suave-geth attach /tmp/geth.ipc
```

► IPC not found

From within the console, you can run:

```
bash eth.accounts[0]
```

This should return "0xb5feafbdd752ad52afb7e1bd2e40432a485bbb7f" - the default funded account for local development.

```
bash eth.getBalance(eth.accounts[0])
```

Should return a really large number 1.1579...e+77. If you try `eth.getBalance("<your_new_address>")` instead, you should see 0.

If you try:

```
bash web3.eth.blockNumber
```

It should tell you the block height of your local network.

What am I actually running?

The main actor in the SUAVE protocol is called a "Kettle". Kettles house all components necessary to perform confidential compute.

Here is the architecture of a Kettle on the Rigil Testnet. When you start a local SUAVE devnet, you're running all the stuff in the purple square (but not the domain specific services, i.e. nodes connected to other chains from/to which you wish to receive or send bundles).

You can read more about exactly what a Kettle contains in [architecture section of the Technical Specs](#)